

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2002-135248

(43)Date of publication of application : 10.05.2002

(51)Int.Cl.

H04L 12/22

G06F 13/00

H04L 12/24

H04L 12/26

(21)Application number : 2000-320008

(71)Applicant : MIZOGUCHI FUMIO
KUREO:KK

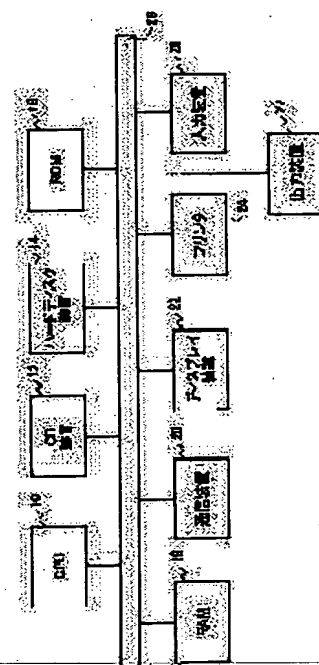
(22)Date of filing : 19.10.2000

(72)Inventor : MIZOGUCHI FUMIO

(54) NETWORK-MONITORING METHOD, NETWORK-MONITORING SYSTEM AND STORAGE MEDIUM RECORDING ITS PROGRAM

(57)Abstract:

PROBLEM TO BE SOLVED: To provide a network-monitoring method and a network-monitoring system which can monitor the state of application of a network, detect unauthorized access in a network or its apprehension, and inform abnormality to a system manager on real time, and storage medium recording its program.
SOLUTION: User' events to a network are collected as log data. From the collected log data, a profile showing the state of ordinary application of the network is formed for each user. Newly collected log data are compared with the profile for each user. By displaying compared results of the newly collected log data, the state of application of the network can be monitored from the compared results of the newly collected log data which are displayed. Thereby the unauthorized access in the network or its apprehension can be detected, and the abnormality can be informed on real time.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

(19) 日本国特許庁 (J P)

(12) 公 開 特 許 公 報 (A)

(11) 特許出願公開番号
特開2002-135248
(P2002-135248A)

(43) 公開日 平成14年5月10日 (2002.5.10)

(51) Int.Cl. ⁷	識別記号	F I	テームト* (参考)
H 0 4 L 12/22		G 0 6 F 13/00	3 5 1 N 5 B 0 8 9
G 0 6 F 13/00	3 5 1	H 0 4 L 11/26	5 K 0 3 0
H 0 4 L 12/24		11/08	
12/26			

審査請求 未請求 請求項の数7 O L (全 13 頁)

(21) 出願番号 特願2000-320008(P2000-320008)

(22) 出願日 平成12年10月19日 (2000. 10. 19)

(71) 出願人 597172890

溝口 文雄

東京都目黒区目黒1-17-3

(71) 出願人 595155484

株式会社クレオ

東京都港区高輪3丁目19番22号

(72) 発明者 溝口 文雄

東京都目黒区目黒1-17-3

(74) 代理人 100070150

弁理士 伊東 忠彦

Fターム(参考) 5B089 GA21 GB02 JB16 KA17 LB14

5K030 GA15 HA05 JA10 KX30 MA04

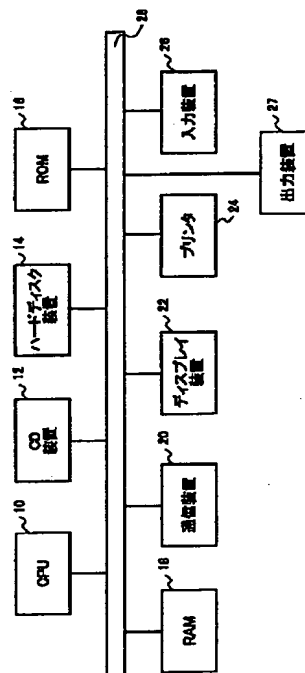
MC09

(54) 【発明の名称】 ネットワーク監視方法、ネットワーク監視システム及びそのプログラムを記録した記録媒体

(57) 【要約】

【課題】 本発明は、ネットワーク使用状況を監視してネットワーク不正侵入またはそのおそれを検知し、リアルタイムにシステム管理者に異常を知らせることのできるネットワーク監視方法、ネットワーク監視システム及びそのプログラムを記録した記録媒体を提供することを目的とする。

【解決手段】 ネットワークに対するユーザのイベントをログデータとして収集し、収集されたログデータからユーザ毎に通常のネットワーク使用状況を表すプロファイルを生成し、新たに収集したログデータをプロファイルとユーザ毎に照合し、新たに収集したログデータの照合結果を表示することにより、表示される新たに収集したログデータの照合結果からネットワーク使用状況を監視でき、ネットワーク不正侵入またはそのおそれを検知することができ、リアルタイムに異常を知らせることができる。



【特許請求の範囲】

【請求項 1】 ネットワークに対するユーザのイベントをログデータとして収集し、
収集されたログデータからユーザ毎に通常のネットワーク使用状況を表すプロファイルを生成し、
新たに収集したログデータを前記プロファイルとユーザ毎に照合し、
前記新たに収集したログデータの照合結果を表示することを特徴とするネットワーク監視方法。

【請求項 2】 請求項 1 記載のネットワーク監視方法において、
前記新たに収集したログデータの照合結果に応じて前記ログデータの表示形態を異ならせて表示することを特徴とするネットワーク監視方法。

【請求項 3】 ネットワークに対するユーザのイベントをログデータとして収集してデータベースに格納する収集手段と、
前記データベースに格納されているログデータからユーザ毎に通常のネットワーク使用状況を表すプロファイルを生成するプロファイル生成手段と、
前記収集手段で新たに収集したログデータを前記プロファイルとユーザ毎に照合する照合手段と、
前記新たに収集したログデータの照合結果を表示する表示手段とを有することを特徴とするネットワーク監視システム。

【請求項 4】 請求項 3 記載のネットワーク監視システムにおいて、
前記表示手段は、前記新たに収集したログデータの照合結果に応じて前記ログデータの表示形態を異ならせることを特徴とするネットワーク監視システム。

【請求項 5】 請求項 3 または 4 記載のネットワーク監視システムにおいて、
前記表示手段は、収集したログデータを、ユーザ毎に、かつ、時間の推移に応じて 2 次元的に表示することを特徴とするネットワーク監視システム。

【請求項 6】 請求項 3 乃至 5 のいずれか記載のネットワーク監視システムにおいて、
前記プロファイル生成手段は、パラメータ自動調整機能を持つ I L P システムで構成されることを特徴とするネットワーク監視システム。

【請求項 7】 コンピュータを、
ネットワークに対するユーザのイベントをログデータとして収集してデータベースに格納する収集手段と、
前記データベースに格納されているログデータからユーザ毎に通常のネットワーク使用状況を表すプロファイルを生成するプロファイル生成手段と、
前記収集手段で新たに収集したログデータを前記プロファイルとユーザ毎に照合する照合手段と、
前記新たに収集したログデータの照合結果を表示する表示手段として機能させるためのプログラムを記録したコ

ンピュータ読み取り可能な記録媒体。

【発明の詳細な説明】**【0001】**

【発明の属する技術分野】本発明は、ネットワーク監視方法、ネットワーク監視システム及びそのプログラムを記録した記録媒体に関し、特に、ネットワークのアクセス状況を監視してネットワーク不正侵入等を検知するネットワーク監視方法、ネットワーク監視システム及びそのプログラムを記録した記録媒体に関する。

【0002】

【従来の技術】従来より、ネットワークまたはネットワークの構成機器に対し、ネットワークの外部または内部から不正にアクセスするネットワーク不正侵入は社会的問題となっており、ネットワーク不正侵入を検知できる検知システムの登場が望まれている。上記のネットワーク不正侵入は年々増加しており、近年の調査によると、外部犯行と共に、内部犯行によるネットワーク不正侵入の被害件数も無視できないくらい多い。このため、侵入検知への期待が高まっている。

【0003】従来より、統計や機械学習を利用してシステムの正常な運用状態を把握し、例外を検出することにより異常検知 (anomaly detection) を行うという研究がある。この研究は、電話の詐欺を検知するために普段の通話記録を機械学習によりルール化し、モニタリングするというものである。

【0004】また、ユーザのコマンドログに対してページアンネットを利用して行動を分析する研究や、機械学習によりコマンドログのシーケンスを解析する研究などがある。

【0005】

【発明が解決しようとする課題】しかし、既存の研究は未だ検討段階であり、ネットワークシステムへの不正アクセスが行われた段階で、ネットワーク不正侵入を検知してネットワーク管理者に異常を知らせる具体的な技術はなかった。

【0006】本発明は、上記の点に鑑みなされたもので、ネットワーク使用状況を監視してネットワーク不正侵入またはそのおそれを検知し、リアルタイムにシステム管理者に異常を知らせることのできるネットワーク監視方法、ネットワーク監視システム及びそのプログラムを記録した記録媒体を提供することを目的とする。

【0007】

【課題を解決するための手段】請求項 1 に記載の発明は、ネットワークに対するユーザのイベントをログデータとして収集し、収集されたログデータからユーザ毎に通常のネットワーク使用状況を表すプロファイルを生成し、新たに収集したログデータを前記プロファイルとユーザ毎に照合し、前記新たに収集したログデータの照合結果を表示することにより、表示される新たに収集したログデータの照合結果からネットワーク使用状況を監視

でき、ネットワーク不正侵入またはそのおそれを検知することができ、リアルタイムに異常を知らせることができる。

【0008】請求項2に記載の発明は、請求項1記載のネットワーク監視方法において、前記新たに収集したログデータの照合結果に応じて前記ログデータの表示形態を異ならせて表示することにより、ログデータの表示形態から視覚的にネットワーク使用状況をリアルタイムに監視できネットワーク不正侵入を検知することができる。

【0009】請求項3に記載の発明は、ネットワークに対するユーザのイベントをログデータとして収集してデータベースに格納する収集手段と、前記データベースに格納されているログデータからユーザ毎に通常のネットワーク使用状況を表すプロファイル生成するプロファイル生成手段と、前記収集手段で新たに収集したログデータを前記プロファイルとユーザ毎に照合する照合手段と、前記新たに収集したログデータの照合結果を表示する表示手段とを有することにより、表示される新たに収集したログデータの照合結果からネットワーク使用状況を監視でき、ネットワーク不正侵入またはそのおそれを検知することができ、リアルタイムに異常を知らせることができる。

【0010】請求項4に記載の発明は、請求項3記載のネットワーク監視システムにおいて、前記表示手段は、前記新たに収集したログデータの照合結果に応じて前記ログデータの表示形態を異ならせることにより、ログデータの表示形態から視覚的にネットワーク使用状況をリアルタイムに監視でき、ネットワーク不正侵入として検知されたログデータのチェックを容易に行うことができる。

【0011】請求項5に記載の発明は、請求項3または4記載のネットワーク監視システムにおいて、前記表示手段は、収集したログデータを、ユーザ毎に、かつ、時間の推移に応じて2次元的に表示することにより、大量のログデータの全貌を表示することが可能となる。

【0012】請求項6に記載の発明は、請求項3乃至5のいずれか記載のネットワーク監視システムにおいて、前記プロファイル生成手段は、パラメータ自動調整機能を持つILPシステムで構成されることにより、最良の性能パラメータ値を示すエラー率を用いて学習した結果の集合をプロファイルとして得ることができる。

【0013】請求項7に記載の発明は、コンピュータを、ネットワークに対するユーザのイベントをログデータとして収集してデータベースに格納する収集手段と、前記データベースに格納されているログデータからユーザ毎に通常のネットワーク使用状況を表すプロファイル生成するプロファイル生成手段と、前記収集手段で新たに収集したログデータを前記プロファイルとユーザ毎に照合する照合手段と、前記新たに収集したログデータ

の照合結果を表示する表示手段として機能させるためのプログラムを記録したコンピュータ読み取り可能な記録媒体を用いることにより、表示される新たに収集したログデータの照合結果からネットワーク使用状況を監視でき、ネットワーク不正侵入またはそのおそれを検知することができ、リアルタイムに異常を知らせることができる。

【0014】

【発明の実施の形態】図1は、本発明のネットワーク監視システムが構築されるコンピュータシステムの一実施例のブロック構成図を示す。同図中、CPU10はシステム全体の制御を司る。CD（コンパクトディスク）装置12にセットされて再生されるCDには、CPU10で実行されるログチェッカー、プロファイリングエンジン、ビジュアルブラウザ等のプログラムが格納されている。ハードディスク装置14にはブートストラッププログラムやシステムプログラムが格納される他に、ログデータベースが格納される。また、ROM16には立ち上げ時にハードディスク装置14からブートストラッププログラムやシステムプログラムを読み込むためのプログラムが格納されている。

【0015】RAM18は、CPU10が実行するプログラムを一時記憶したり、演算結果を記憶するための作業領域として使用される。通信装置20はLANやインターネット等のネットワークに接続する。ディスプレイ装置22はディスプレイ表示を行い、プリンタ24は処理結果等をプリントアウトする。また、入力装置26としてコマンド等を入力するキーボードと、表示画面におけるカーソル移動やクリックを行うためのマウス等が設けられ、出力装置27として発音のためのスピーカが設けられている。上記の各装置間はシステムバス28により接続されている。

【0016】図2は、本発明のネットワーク監視システムのシステムアーキテクチャの一実施例のブロック図を示す。本発明システムは、ログチェッカー30、ログデータベース32、プロファイリングエンジン34、及びビジュアルブラウザ36から成る。ログチェッカー30は、通信装置20が接続されたネットワーク上を流れるユーザイベントを常時モニタリングして、そのユーザイベントをログデータとして収集しログデータベース32に格納する。ログデータベース32に収集されたログデータはプロファイリングエンジン34、及びビジュアルブラウザ36によって参照される。

【0017】プロファイリングエンジン34は、各ユーザの通常の振る舞いを表すプロファイル生成する。このプロファイルの生成には、帰納論理プログラミング（ILP: Inductive Logic Programming）を利用する。生成されたプロファイルは、ログチェッカー30およびビジュアルブラウザ36に送られる。

【0018】ログチェッカー30はこのプロファイルを利用し、ユーザのイベントをプロファイルと照合する。プロファイルに当てはまらないイベントは、そのユーザの通常の利用とは異なるため、不正なイベント即ちネットワーク不正侵入として検知され、検知情報がビジュアルブラウザ36に通知される。

【0019】ビジュアルブラウザ36は、ログデータベース32からログデータを取得し、ログデータそのものを表示すると共に、通常のイベントのログか、もしくは、不正なイベントのログかの分類を表示形態を変えることにより視覚的に表現する。また、システム管理者はビジュアルブラウザ36を利用してイベントやプロファイルを編集することが可能である。これは、不正なイベントだけを検知するための完全なプロファイルの生成が困難であるため、システム管理者がイベントまたはプロファイルを編集するために必要となる。

【0020】現実的には、不正なイベントは必ず検知するが、まれに正常なイベントを不正なイベントとして検知する程度のプロファイルをプロファイリングエンジン34で生成する。この場合、システム管理者はネットワーク侵入等の不正なイベントとして検知されたイベントが本当に不正なものかどうかをチェックする必要があるが、ビジュアルブラウザ36ではログデータを視覚的に表現するため、上記チェック作業が容易になる。ここで、対象とするユーザのイベントのログデータとしては、ユーザ名(who)、マシン名(where)、イベント名(what)、イベント引数(how)、イベント実行時刻(when)が上げられる。これは、どのユーザが、いつ、どこで、何を、どのように実行したかを表すものである。なお、ログデータとして、この5つの情報は必要不可欠ではなく、いずれかの情報を含んでいれば利用可能である。

【0021】ビジュアルブラウザ36は、ログデータの効率的な閲覧を可能とし、システム管理者に対して、視覚的にログのデータ構造を示すもので、通常のユーザの利用状況の把握や、ログチェッカー30がネットワーク不正侵入を検知した場合に、どのようなイベントを実行したかの痕跡を発見し易くする。ビジュアルブラウザ36は、ディスプレイ装置22に、図3に示すログブラウザパネル、図4に示すプロパティパネル、図5に示すビ
40 ジュアライズパネル、図6に示すプロファイルパネル、それぞれを表示する。

【0022】図3は、ログブラウザパネルの一実施例を示す。同図中、ログ表示部40に複数のログを行単位で表示する。また、テキストフィールド42に、ログデータベース32からログを取得する際の検索条件を記入する。この実施例では、ユーザ名「hiraishi」かつ、ホスト名「imct451」の検索条件を満足するログデータの取得することを指示している。テキストフ
50 イールド42に記入された「&」は論理積を表すが、

「&」の代わりに「|」を用いて論理和を指示することも可能である。

【0023】Getボタン43は、テキストフィールド42に記入した条件によるログデータの取得を指示する。また、Clearボタン44は、テキストフィールド42のクリアを指示する。Backボタン45はGetボタン43をクリックする一つ前に表示したログデータをログ表示部40表示させることを指示する。ところで、テキストフィールド42に未記入の状態でGetボタン43をクリックすると、後述するプロパティパネルで設定された条件に相当するログデータが取得されてログ表示部40に表示される。なお、テキストフィールド42、Getボタン43、Clearボタン44、Backボタン45は、ログブラウザパネルだけでなくプロパティパネル、ビジュアライズパネル、プロファイルパネルにも共通に表示される。

【0024】ログ表示部40には、ログデータベース32から取得した複数のログを行単位で表示される。各行左端の第1カラムにはユーザ名、第2カラムにはホスト名、第3カラムには実行したコマンド及びその引数、右端の第4カラムには実行した日付と時刻が表示される。また、ログ表示部40に表示されているユーザ名やホスト名をマウスで選択してクリックすると、選択されたユーザ名やホスト名がテキストフィールド42にコピーされ、自動的に項目(ユーザ名やホスト名)に関連するログデータがデータベース32から取得される。このようにして、マウス操作によるログのブラウジングを可能としている。

【0025】ところで、ログチェッカー30によって、不正なイベントの通知があった場合には、ビジュアルブラウザ36はカッコーアイコン46が動かし、カッコーの鳴き声をスピーカより発音することによって、不正なイベントが発生したことをシステム管理者に通知する。また、不正なイベントに対応するログデータを例えば赤い文字で表示する。このように、視覚と聴覚によって、システム管理者に対し不正なイベント発生
50 の通知が行われる。

【0026】図4は、プロパティパネルの一実施例を示す。同図中、指定欄50ではログデータを取得する期間(デフォルト値)を指定する。指定欄52では、ログ表示部40に表示されているユーザ名やホスト名等の項目をクリックしてテキストフィールド42にコピーする際に、複数の項目の論理積と論理和のいずれを用いるかを設定する。指定欄54ではログデータを取得するタイミングを指定する。また、指定欄56ではログチェッカー30との通信を行うためにログチェッカー30のIPアドレス及びポート番号を指定する。

【0027】同様に、プロファイリングエンジン34のIPアドレス及びポート番号を指定するための図示されていない欄も設けられている。上記のログチェッカー30

やプロファイリングエンジン34のIPアドレス及びポート番号の指定は、ログチェッカー30やプロファイリングエンジン34がビジュアルブラウザ36とは異なるコンピュータで動作する場合に必要なものであつて、図1に示すように、ログチェッカー30やプロファイリングエンジン34がビジュアルブラウザ36と同一のCPUで動作する場合には必要ない。

【0028】図5は、ビジュアライズパネルの一実施例を示す。ビジュアライズパネルの最も重要な役割はマシンやネットワークの状態を容易に理解させることにある。図5において、ワイヤフレームによる楕円球60は、縦軸にユーザ名及びマシン名からなる属性が表現され、横軸の左から右に時間の推移が表現された2次元構成である。縦横の線で区画された各セルには、ユーザが実行したコマンド（ログデータのコマンド）が表示される。

【0029】楕円球60の中央に向かうほどセルの表示領域は大きく、外枠に近くなるにつれてセルの表示領域は小さくなる。このため、外枠に近づくにつれて表示は小さくなるが、一度に大量のログデータの全貌を表示することが可能である。また、任意のセルをマウスでドラッグすることによって、楕円球60を上下左右に回転するようにスクロールさせることによって、目的の属性（ユーザ及びマシン）及び時間の推移に沿ってログデータの内容を見ることが可能である。この時、楕円球60の中央に位置するログデータの属性が左端部に選択ログ属性62として表示され、また、楕円球60の中央に位置するログデータのコマンドの実行時刻が下端部にコマンド実行時刻64として表示される。

【0030】また、新たなログデータが得られた場合には、最新のログデータを楕円球60の中央のセル60aに表示するように、自動的にスクロールが行われる。図5に示す例は、楕円球60は中央のセル60aに表示されているログデータ「ls」が得られスクロールされた状態を表している。

【0031】更に、大量のログデータの中で、不正なイベントのログデータの発見を容易にするため、ログデータのコマンドに対して以下の4つのカテゴリに分類を行い、カテゴリによって表示するセルの色分けを行う。

【0032】第1のカテゴリは、Un-ruledコマンドであり赤で表示する。このカテゴリは最も危険なカテゴリであり、当該属性のプロファイルに適合しないコマンド群である。つまり、このカテゴリに含まれるコマンドは、ユーザが本来実行しないコマンドであるため、そのユーザになりすましたネットワーク不正侵入者によって、実行されたコマンドである可能性が高い。

【0033】第2のカテゴリは、Dangerコマンドであり黄で表示する。このカテゴリには、ネットワークに関連するコマンドや、システム管理者用のコマンドが属し、例えばsu, finger, ftp, telne

t, rlogin, alias, configure, nmap等のコマンドである。これらのコマンドは、ネットワークへの不正侵入や、パスワードデータを盗み見るために利用されるものである。

【0034】第3のカテゴリは、Safetyコマンドであり青で表示する。このカテゴリは、当該属性のプロファイルに適合するコマンドや間違いなく安全なコマンドがこのカテゴリに含まれ、例えばls, cd, latex, java等のコマンドである。

【0035】第4のカテゴリは、Unknownコマンドであり緑で表示する。このカテゴリは上記第1～第3のカテゴリに属さないコマンド群である。これは、多くの場合タイプミスされたコマンドや、ユーザが独自に作成したマクロのコマンドが含まれる。この第4のカテゴリも、ネットワーク不正侵入者によって作成されたコマンドの可能性は高い。

【0036】大量のログデータを図5の楕円球60に表示した場合、外枠に近いものの内容の表示は不可能であるが、色の表示は可能である。したがって、上記のようにカテゴリ毎にセルの色分けを行えば、色の表示をみるだけで不正なイベントを発見することが可能である。さらに、発見したセルをドラッグして楕円球60の中央位置に移動させることでログデータの内容を読み取ることができ、何が実行されたのかを把握することが可能である。

【0037】例えば、システム管理者は、重要度の高い順に、例えば、赤色、黄色、緑色のセルに注目し、その色のセルが存在する場合には、その内容のチェックを行う。これによって、システム管理者は大量なログデータを視覚的に理解し、その内容を確認することが可能となる。

【0038】なお、カテゴリに応じて色を異ならせる以外に、文字のフォントを異ならせたり、点滅間隔を異ならせる等の、他の表示形態を異ならせる方法を採用しても良い。

【0039】図6は、プロファイルパネルの一実施例を示す。同図中、プロファイル表示部70には、プロファイリングエンジン34によって生成されたプロファイルがユーザ毎にリストとして表示される。このリストはプロファイルを構成するルールを行毎に表示している。

【0040】図6に示す例では、第1行の記述「-----hiraishi-----」がユーザ名を表している。リストの第2行の記述「*, java,」における「*」は全てに当てはまることを示すワイルドカードであり、このユーザは「java」というコマンドをマシンに関係なく利用するというルールである。以降の行にそのユーザの通常のコマンドのルールが続いて表示されている。第5行の記述は、このユーザはマシン名「imctr03」で、コマンド「kterm」を実行し、そのとき、引数の一番目に「&」を指定するというルール

である。第2行以降の各ルールは論理和の関係となる。

【0041】プロファイリングエンジン34によって生成されるプロファイルは完全ではないため、プロファイル表示部70では、各行のルールを編集したり、新たなルールの追加やルールの削除が可能である。また、ボタン72をクリックすることで全てのユーザについての新たなプロファイルのプロファイリングエンジン34から取得する。また、欄74によって、新たなプロファイルのプロファイリングエンジン34から取得するタイミング（図示の例では60秒毎に取得）を設定することができる。

【0042】プロファイリングエンジン34では帰納論理プログラミング（ILP）システムを利用する。ILPシステムは、機械学習の一つの手法であり、一階述語論理に基づいた例からの関係学習、更に、逐次的な処理が可能なシステムである。そして、ILPは次のようなフレームワークに基づいている。正事例 E^+ 、負事例 E^- 、それらを説明する背景知識 B を与えたとき、（1）式により正事例を含み、（2）式により負事例を排除する仮説 H を仮説探索により見つける。

【0043】

【数1】

$$B \wedge H \models E^+ \quad \dots(1)$$

$$B \wedge H \wedge E^- \neq \phi \quad \dots(2)$$

ここで、記号 \models は伴意と呼び(1)式は $B \wedge H$ から E^+ が導かれることを意味する。

ILPでは、ユーザが与えた宣言情報を利用して述語を結合する。これは、規則にのっとって関係データベース内の複数テーブルを結合するのと等しい。これは、従って、このテーブルに複数のログデータを格納し、そこから規則を導出する。本実施例では汎用性を考え、そのILPシステムをJava言語によって実装する。

【0044】ここでは、仮説生成を行なうとき、入出力モードと変数結合の深さを利用して、最も特殊な節をまず求める。そして、最も一般的な仮説から最も特殊な仮説（ボトム節）に向けて仮説を徐々に特殊化する。この時、仮説における正事例の包含率と仮説の長さを目的関数とし、含んでもよい負事例の割合（エラー率）を制約条件として与え、この目的関数が最大となる仮説を最良仮説、即ちルールとする。このように、ILPシステムは、学習時に負事例をどの程度含んでもよいかというエラー率を設定しなければならない。その設定次第でルールの性能が左右される。

【0045】ここで、得られたルールが正事例と負事例をどの程度正しく説明できるかを示す精度（Accuracy）、正と判別する規則が正事例をどの程度判別できるかを示す感度（Sensitivity）、負と判別する規則が負事例をどの程度判別できるかを示す特殊性（Specificity）の3種類の性能パラメータ

を次のように定義する。

【0046】正事例 E^+ が仮説 H で正として判断するTP（True Positive）と仮説 H で負として判断するFN（False Negative）とからなり、負事例 E^- が仮説 H で正として判断するFP（False Positive）と仮説 H で負として判断するTN（True Negative）とからなるとき、性能パラメータは次の式で表される。

$$\text{【0047】感度} = TP / E^+$$

$$\text{特殊性} = TN / E^-$$

$$\text{精度} = (TP + TN) / (E^+ + E^-)$$

これらの性能パラメータは、リサンプリング手法により求まる。そして、指定した性能パラメータを最大化し、指定されなかった残りの性能パラメータを制約条件として定義する。この時、性能パラメータの特徴として、エラー率を大きくすると感度はよくなり精度は悪くなる傾向がある。そこで、エラー率と各性能パラメータの関係を考慮に入れた探索により、最良の性能パラメータ値を示すエラー率を求める。その時のエラー率を用いて学習した結果の集合をプロファイルとする。

【0048】図7は、プロファイリングエンジン34内のILPシステムが実行するエラー率自動調整処理の一実施例のフローチャートを示す。

【0049】同図中、ステップS10でシステム管理者は最大化したい性能パラメータを決め、制約条件として残り2つの性能パラメータ値を与える。ステップS12で停止条件を満足したか否かを判別して、停止条件を満足しない場合は、ステップS14に進む。ここではクロスバリデーションやブートストラップのようなリサンプリング処理により訓練集合とテスト集合を作成する。

【0050】次に、ステップS16でILPシステムは訓練集合を学習し、それをテストして3つの性能パラメータを求める。そして、ステップS18でシステム管理者が指定した制約条件を2つの性能パラメータが満足しているかどうかを確認する。ここで、満足している場合は、ステップS20の調整プロセスに進んで、最大化したい性能パラメータが最大になるようにエラー率を更新してステップS12に進みステップS12～S18を繰り返す。

【0051】一方、ステップS18で制約条件を満足している場合には、ステップS22の最大化プロセスに進んで最大化したい性能パラメータが最大になるようにエラー率を更新してステップS12に進みステップS12～S18を繰り返す。そして、最大化したい性能パラメータの値に変動がなくなると、ステップS12で停止条件を満足してこの処理を終了する。

【0052】上記の処理によって、ログデータベース32のログイン状況とコマンドログを取得することにより、例えば次に示すようなルールが作成される。

【0053】user_A(X) : -login

(X, ' ' imctX' ', Y), type (Y, ' ' lpr, pimc3, 3' ') .

このルールは、ホスト名imctXではユーザAはプリンタ利用のコマンドlprをタイプし、そのとき第3引数にネットワークプリンタを指定するpimc3を用いることを表している。このようなルールの集合をあるカテゴリのプロファイル集合とする。

【0054】ログデータベース32では、ファイルと関係データベースとを利用してログデータの管理を行う。関係データベースは例えば、「ID(識別番号)、ユーザ名、ホスト名、タイプされたコマンド(引数を含む)」、タイプされたコマンド(引数を含まない)、タイプされた時刻」のテーブルと、「ID、タイプされたコマンド(引数を含む)、コマンド引数」のテーブルとの2つのテーブルから構成する。また、ファイルは例えば、上記関係データベースの各テーブルを1つのファイルとして、その中にCSV(Comma Separated Value)形式で上記関係データベースと同一データを格納する。

【0055】このように、ファイルと関係データベースを用いるのは、関係データベースが利用できない環境であっても、ファイルを利用できるようにしておくためであり、ファイルと関係データベースのいずれか一方だけを利用するものであっても良い。

【0056】関係データベース、ファイル共に、関係データベースを演算するための言語であるSQL(Structured Query Language)を用いて演算を行い、ログデータの加工と検索を行なう。図8に示すように、ビジュアルブラウザ36のログブラウザパネル、プロパティパネル、ビジュアルライズパネル、プロファイルパネルそれぞれは、必要な情報をログデータベース32から取得するときに、演算要求「クエリ」をログデータベース32に供給して、ログデータの加工及び検索を行い、その演算結果を得て、上記の各パネルに表示する。また、プロファイリングエンジン34は、プロファイルを作成する際に関係抽出や仮説の包含数を導出するときに、そのための演算要求をログデータベース32に供給してログデータベース32を利用する。ログチェッカー30は、分散型のログ収集を行なう。この場合、ログチェッカー30は、図9に示すように、ログ収集サーバ80と複数のモニタエージェント82a~82xとからなる。対象としたいログ(ユーザ名やホスト名)が決まれば、対象のログを監視するモニタエージェントを作成し、それにログを監視させる。ここでいう監視とは、ログの収集及びプロファイルとの照合による検知である。

【0057】ログの収集では、各モニタエージェント82a~82xが収集したログをログ収集サーバ80から例えばTCP/IP接続を利用してログデータベース32の対象ファイル、または、対象関係データベースに付

け加える。プロファイルとの照合による検知では、各モニタエージェント82a~82xは、プロファイリングエンジン34で作成されたプロファイルと得られたログを照合し、そのログが異常の有無を示すフラグ等の検知情報をビジュアルブラウザ36に供給する。

【0058】図10は、ログチェッカー30が実行する処理の一実施例のフローチャートを示す。同図中、ステップS30でネットワーク上を流れるログデータを取得する。次に、ステップS32で取得したログデータをログデータベース32に送信する。ステップS34では取得したログデータをプロファイリングエンジン34から供給されている対応するプロファイルと照合して、異常なログか否かを判別する。ここで、異常と判別されるとステップS36で異常なログデータをビジュアルブラウザ36に送信してステップS30に進み、異常がないと判別されるとそのままステップS30に進んで、上記の処理を繰り返す。

【0059】図11は、ログデータベース32が実行する処理の一実施例のフローチャートを示す。同図中、ステップS40でログチェッカー30から送信されたログデータを受信し、ステップS42でこのログデータを保存する。次に、ステップS44でプロファイリングエンジン34からログ取得要求があるか否かを判別する。ログ取得要求があった場合はステップS46でログ取得要求があったログデータをプロファイリングエンジン34に送信してステップS48に進む。ログ取得要求がなかった場合はそのままステップS48に進む。

【0060】ステップS48ではビジュアルブラウザ36からログ取得要求があるか否かを判別する。ログ取得要求があった場合はステップS49でログ取得要求があったログデータをビジュアルブラウザ36に送信してステップS40に進む。ログ取得要求がなかった場合はそのままステップS40に進んで、上記の処理を繰り返す。

【0061】図12は、ビジュアルブラウザ36が実行する処理の一実施例のフローチャートを示す。同図中、ステップS50でログチェッカー30またはログデータベース32から送信されたログデータ、またはプロファイリングエンジン34から送信されたプロファイルを受信する。次に、ステップS52で受信したログデータは異常なログであるか否かを判別する。異常なログであればステップS54で警報を鳴らし、ステップS56に進む。異常なログでなければそのままステップS56に進む。ステップS56ではログデータの分類及び表示を行い、ステップS50に進んで、上記の処理を繰り返す。

【0062】図13は、ステップS56で実行されるログデータの分類及び表示処理の一実施例のフローチャートを示す。同図中、ステップS60でログチェッカー30から送信されたログデータのコマンドを、そのログデータのユーザ名に対応するプロファイルと照合して照合

できたか否かを判別する。照合できなかったログデータのコマンドについてはUnruledコマンドであるので、ステップS62に進んで赤で表示を行う。

【0063】照合できたログデータのコマンドについてはステップS64で、su, finger, ftp, telnet, rlogin, alias, configure, nmap等のDangerコマンドと一致するか否かによりDangerコマンドに分類されるか否かを判別し、Dangerコマンドに分類されると、ステップS66に進んで黄で表示を行う。

【0064】Dangerコマンドに分類されないログデータのコマンドについてはステップS68で、ls, cd, jlatex, java等のSafetyコマンドと一致するか否かによりSafetyコマンドに分類されるか否かを判別し、Safetyコマンドに分類されると、ステップS70に進んで青で表示を行う。Safetyコマンドに分類されないログデータのコマンドについてはステップS72でUnknownコマンドに分類して緑で表示を行う。

【0065】図14は、プロファイリングエンジン34が実行する処理の一実施例のフローチャートを示す。同図中、ステップS80でログデータベース32に対し任意のユーザ名に対応する過去の全てのログデータの送信要求を行い、これによりログデータベース32から送信されたログデータを受信する。次に、ステップS82で受信した任意のユーザ名に対応する過去の全てのログデータからプロファイルを作成する。

【0066】この後、ステップS84で作成したプロファイルログチェッカー30に送信し、ステップS86で作成したプロファイルをビジュアルブラウザ36に送信したのち、ステップS80に進んで、上記の処理を繰り返す。

【0067】ここで、ユーザ名「yoshii」に対応して、図15に示すログデータがログデータベース32に格納されているものとする。図15では、各2行で1つのコマンドを表す。第1行の「+0964615658」はコマンドの実行時間であり、「imctut01」は実行マシン名であり、第2行の「jlatex a. tex」は実行コマンドである。この場合、プロファイリングエンジン34は、図15に示すログデータから図16に示すプロファイルを生

成する。【0068】図16に示すユーザ名「yoshii」のプロファイルにおいて、第2行のプロファイル「*, jlatex, *」は、図15の第1, 2行のログデータと第7, 8行のログデータから生成され、両ログデータのマシン名、引数が異なるため、単にjlatexというコマンドを利用する、といった意味のプロファイルである。図16の第3行のプロファイルは、図15の第3, 4行のログデータと第9, 10行のログデータから生成されている。

【0069】図16の第4行のプロファイルは、図15の第5, 6行のログデータと第11, 12行のログデータから生成され、両ログデータともマシン名はimctut02で、1番目の引数に-Pimc3が指定されているので、lprというコマンドをimctut02というマシンで引数の1番目に-Pimc3を利用する、といった意味のプロファイルである。

【0070】上記のプロファイルが得られた後、図17に示すログデータが新たに得られた場合について説明する。図17の第1, 2行のログデータ及び第3, 4行のログデータはプロファイル「*, jlatex, *」に一致するためログチェッカー30では正常と判別される。

【0071】図17の第5, 6行のログデータはプロファイル「*, dvi2ps, *」に一致するためログチェッカー30では正常と判別される。図17の第7, 8行のログデータはプロファイル「imctut02, lpr, -Pimc3, 1」に対し、1番目の引数-Pimc3が異なるためログチェッカー30では異常と判別され、この第7, 8行のログデータはビジュアルブラウザ36に送信される。

【0072】

【発明の効果】上述の如く、請求項1に記載の発明は、ネットワークに対するユーザのイベントをログデータとして収集し、収集されたログデータからユーザ毎に通常のネットワーク使用状況を表すプロファイルを生

成し、新たに収集したログデータを前記プロファイルとユーザ毎に照合し、前記新たに収集したログデータの照合結果を表示することにより、表示される新たに収集したログデータの照合結果からネットワーク使用状況を監視でき、ネットワーク不正侵入またはそのおそれを検知することができる、リアルタイムに異常を知らせることができ

る。

【0073】請求項2に記載の発明は、新たに収集したログデータの照合結果に応じて前記ログデータの表示形態を異ならせて表示することにより、ログデータの表示形態から視覚的にネットワーク使用状況をリアルタイムに監視できネットワーク不正侵入を検知することができ

る。

【0074】請求項3に記載の発明は、ネットワークに対するユーザのイベントをログデータとして収集してデータベースに格納する収集手段と、前記データベースに格納されているログデータからユーザ毎に通常のネットワーク使用状況を表すプロファイルを生

成するプロファイル生成手段と、前記収集手段で新たに収集したログデータを前記プロファイルとユーザ毎に照合する照合手段と、前記新たに収集したログデータの照合結果を表示する表示手段とを有することにより、表示される新たに収集したログデータの照合結果からネットワーク使用状況を監視でき、ネットワーク不正侵入またはそのおそれ

検知することができ、リアルタイムに異常を知らせることができる。

【0075】請求項4に記載の発明では、表示手段は、前記新たに収集したログデータの照合結果に応じて前記ログデータの表示形態を異ならせることにより、ログデータの表示形態から視覚的にネットワーク使用状況をリアルタイムに監視でき、ネットワーク不正侵入として検知されたログデータのチェックを容易に行うことができる。

【0076】請求項5に記載の発明では、表示手段は、収集したログデータを、ユーザ毎に、かつ、時間の推移に応じて2次元的に表示することにより、大量のログデータの全貌を表示することが可能となる。

【0077】請求項6に記載の発明では、プロファイル生成手段は、パラメータ自動調整機能を持つILPシステムで構成されることにより、最良の性能パラメータ値を示すエラー率を用いて学習した結果の集合をプロファイルとして得ることができる。

【0078】請求項7に記載の発明は、コンピュータを、ネットワークに対するユーザのイベントをログデータとして収集してデータベースに格納する収集手段と、前記データベースに格納されているログデータからユーザ毎に通常のネットワーク使用状況を表すプロファイルを生成するプロファイル生成手段と、前記収集手段で新たに収集したログデータを前記プロファイルとユーザ毎に照合する照合手段と、前記新たに収集したログデータの照合結果を表示する表示手段として機能させるためのプログラムを記録したコンピュータ読み取り可能な記録媒体を用いることにより、表示される新たに収集したログデータの照合結果からネットワーク使用状況を監視でき、ネットワーク不正侵入またはそのおそれを検知することができ、リアルタイムに異常を知らせることができる。

【図面の簡単な説明】

【図1】本発明のネットワーク監視システムが構築されるコンピュータシステムの一実施例のブロック構成図である。

【図2】本発明のネットワーク監視システムのシステムアーキテクチャの一実施例のブロック図である。

【図3】ログブラウザパネルの一実施例を示す図である。

【図4】プロパティパネルの一実施例を示す図である。

【図5】ビジュアルライズパネルの一実施例を示す図である。

【図6】プロファイルパネルの一実施例を示す図である。

る。

【図7】プロファイリングエンジン34内のILPシステムが実行するエラー率自動調整処理の一実施例のフローチャートである。

【図8】ビジュアルブラウザ36及びプロファイリングエンジン34によるログデータベース32の利用を説明するための図である。

【図9】分散型のログ収集を行なうログチェッカー30の構成を説明するための図である。

【図10】ログチェッカー30が実行する処理の一実施例のフローチャートである。

【図11】ログデータベース32が実行する処理の一実施例のフローチャートである。

【図12】ビジュアルブラウザ36が実行する処理の一実施例のフローチャートである。

【図13】ステップS56で実行されるログデータの分類及び表示処理の一実施例のフローチャートである。

【図14】プロファイリングエンジン34が実行する処理の一実施例のフローチャートである。

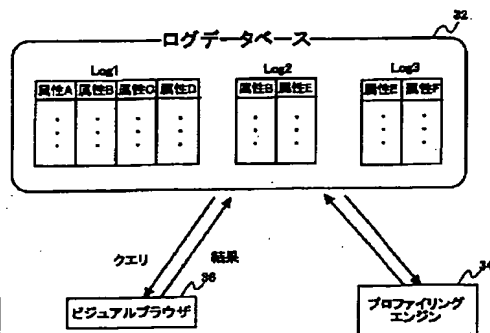
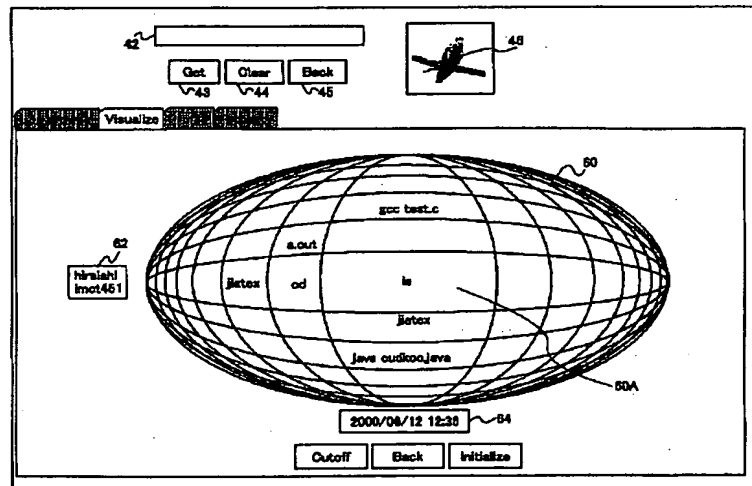
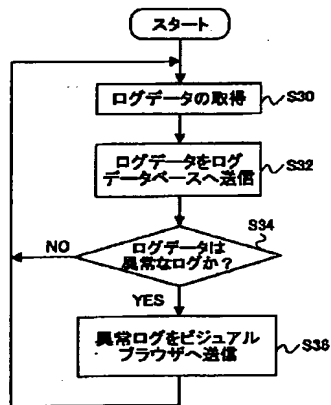
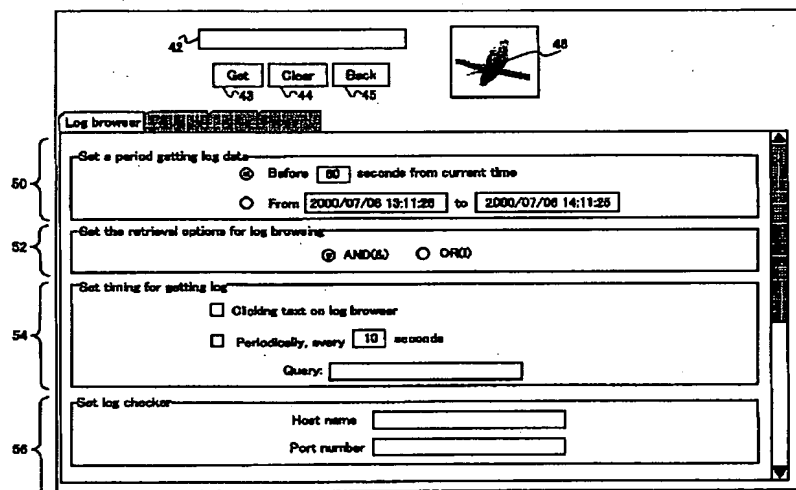
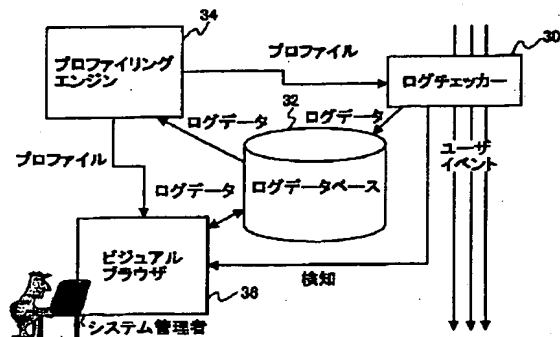
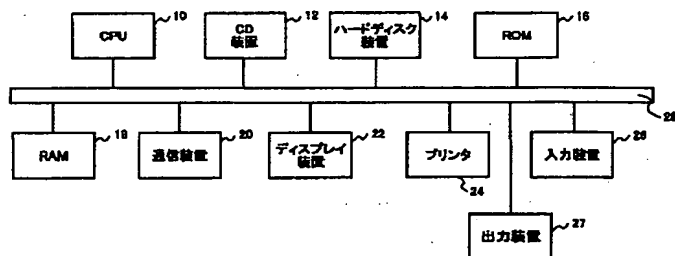
【図15】ログデータベース32に格納されているログデータの一実施例を示す図である。

【図16】生成されるプロファイルの一実施例を示す図である。


【図17】ログデータの一実施例を示す図である。

【符号の説明】

- 10 CPU
- 12 CD装置
- 14 ハードディスク装置
- 20 通信装置
- 22 ディスプレイ装置
- 24 プリンタ
- 26 入力装置
- 27 出力装置
- 30 ログチェッカー
- 32 ログデータベース
- 34 プロファイリングエンジン
- 36 ビジュアルブラウザ
- 40 ログ表示部
- 43 Getボタン
- 44 Clearボタン
- 45 Backボタン
- 46 カッコイアイコン
- 50, 52, 54, 56 指定欄
- 60 楕円球
- 70 プロファイル表示部



【図3】

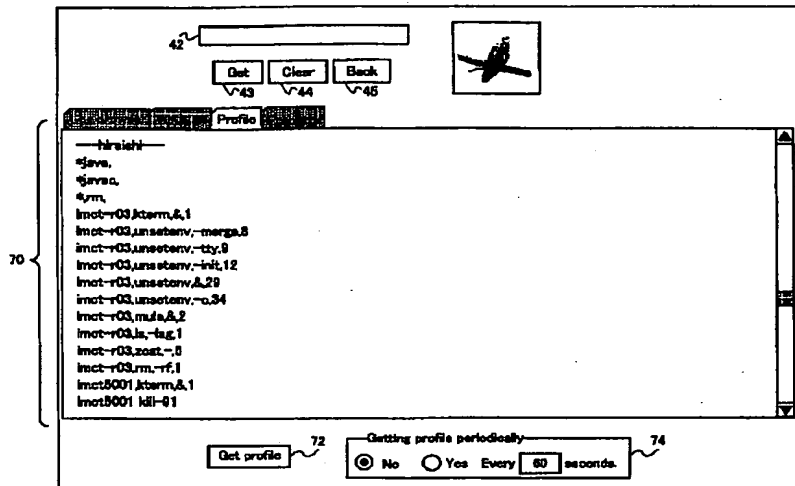
42 46 

43 44 45

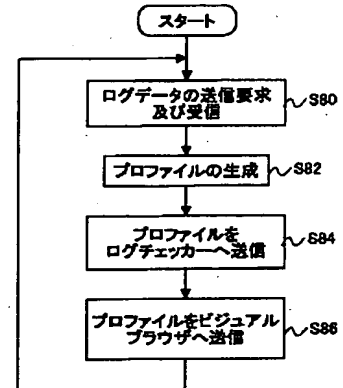
Log browser

User name	Host name	Command	Date/Time
hiraishi	imct451	cd igex	2000/6/12 12:36:19
hiraishi	imct451	ls	2000/6/12 12:36:17
hiraishi	imct451	cd	2000/6/12 12:36:17
hiraishi	imct451	ls	2000/6/12 12:36:18
hiraishi	imct451	latex pria2k.tex	2000/6/12 12:34:31
hiraishi	imct451	latex pria2k.tex	2000/6/12 12:34:23
hiraishi	imct451	latex pria2k.tex	2000/6/12 12:25:52
hiraishi	imct451	latex pria2k.tex	2000/6/12 12:25:51
hiraishi	imct451	latex pria2k.tex	2000/6/12 12:25:49
hiraishi	imct451	lpq-P1mc3	2000/6/11 16:26:37
hiraishi	imct451	adminltool	2000/6/11 16:26:26
hiraishi	imct451	setenv LANG C	2000/6/11 16:26:25
hiraishi	imct451	adminltool	2000/6/11 16:26:15
hiraishi	imct451	lpq-P1mc4	2000/6/11 16:26:11
hiraishi	imct451	lpq-P1mc3	2000/6/11 16:26:05
hiraishi	imct451	lpq-P1mc5	2000/6/11 16:26:02
hiraishi	imct451	lpq-P1mc3	2000/6/11 16:25:55
hiraishi	imct451	ls	2000/6/11 16:25:51
hiraishi	imct451	clear	2000/6/11 16:25:51
hiraishi	imct451	lpq-P1mc3 pria2k.ps	2000/6/11 16:25:49
hiraishi	imct451	ls	2000/6/11 16:25:43

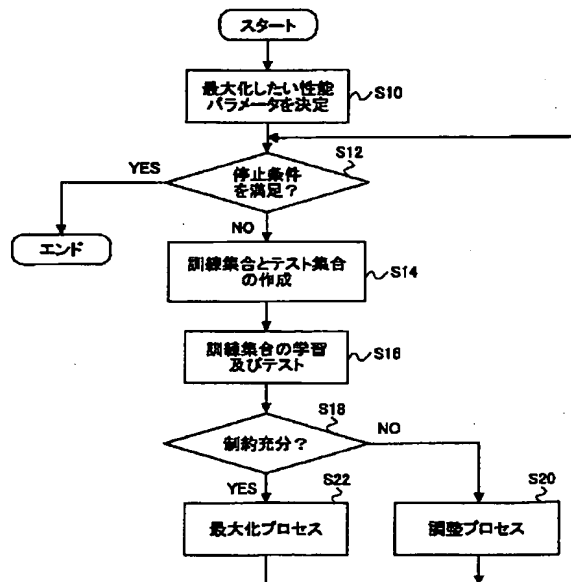
【図6】



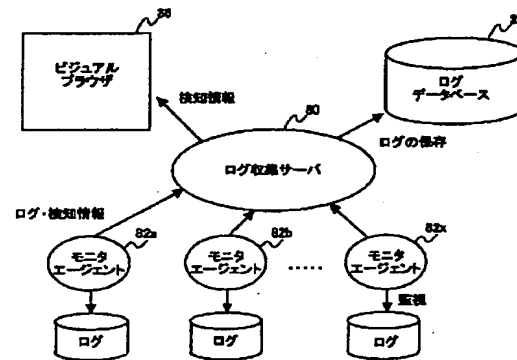
【図14】



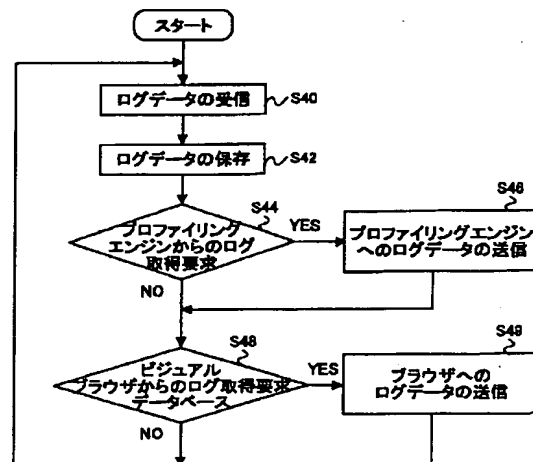
【図7】



【図9】



【図11】



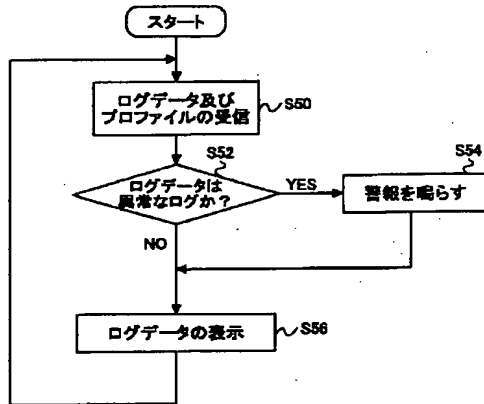
【図15】

行	
1	#+0884815858imctut01
2	jletax a.tex
3	#+0884815860imctut01
4	dvi2ps a.dvi2ps
5	#+0884815878imctut02
6	lpr-Pime3 a.ps
7	#+0884815888imctut02
8	jletax b.tex
9	#+0884815890imctut02
10	dvi2ps-t 1 b.dvi2ps
11	#+0884815708imctut02
12	lpr-Pime3 b.ps

【図16】

行	
1	---yoshih---
2	*jletax.*
3	*dvi2ps.*
4	imctut02.kpr-Pime3.1

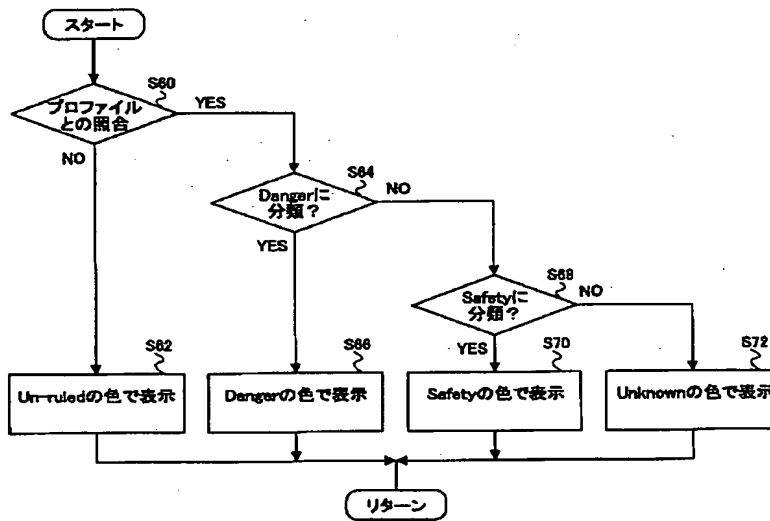
【図12】



【図17】

行		
1	#+0884615858imotut01	正常: プロファイル「+jstex.*」に一致
2	jstex oss2000.tex	
3	#+0884615880imctut01	正常: プロファイル「+jstex.*」に一致
4	jstex oss2000.tex	
5	#+0884615872imotut02	正常: プロファイル「+dvi2ps.*」に一致
6	dvi2ps oss2000.dvi>oss2000.ps	
7	#+0884615876imotut02	異常: プロファイル「imotut02」or「Pimo3.1」マシンのimotut02は合っているが引数が異なる
8	lpr-Pimo5 oss2000.ps	

【図13】



**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☒ FADED TEXT OR DRAWING
- ☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.